**Home Automation using Arduino Cloud**

**Name:** Santiago Cruz

**Date:** 04/10/2024

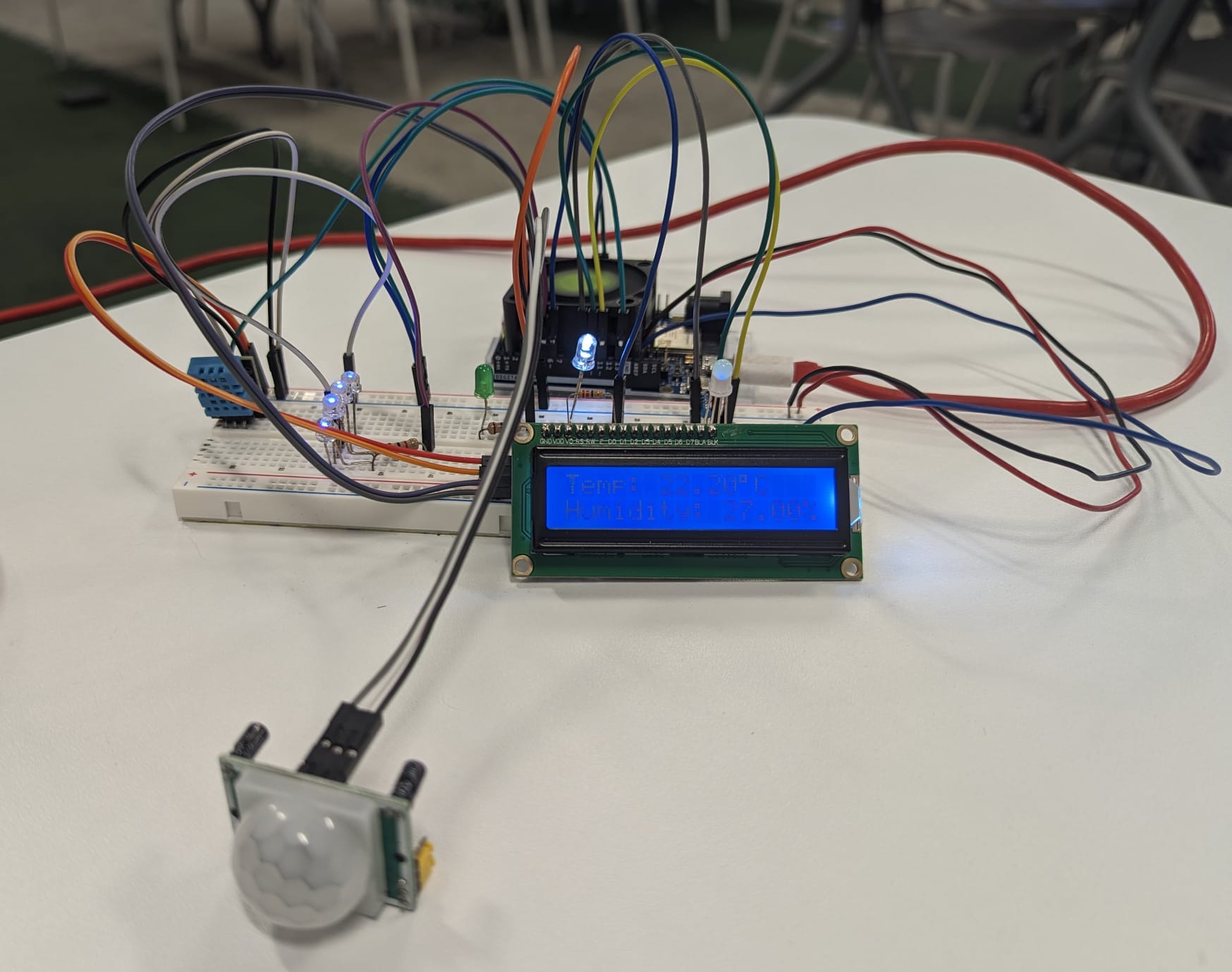
This Home Automation system combines the power of Arduino Cloud with various sensors and LEDs to offer features like motion detection, temperature and humidity monitoring, and customizable lighting control, here are some highlights:

**Enhanced security:** Receive alerts on your phone whenever the PIR sensor detects motion, for improved peace of mind.

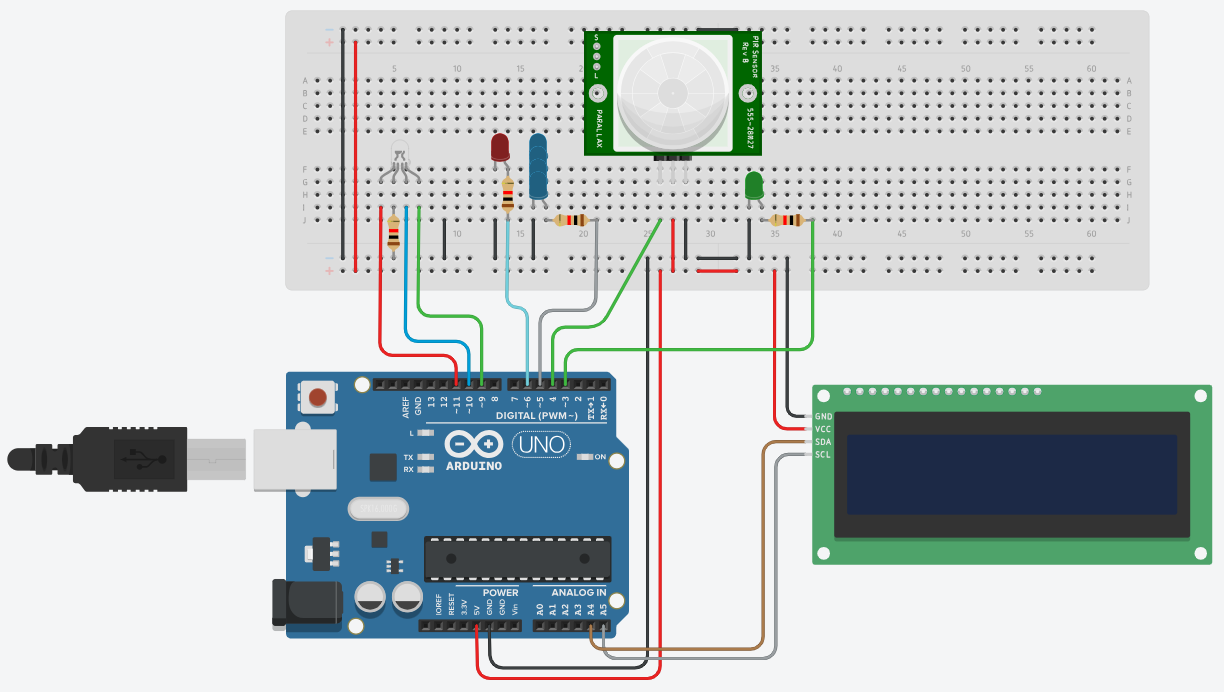
**Real-time monitoring:** Keep an eye on your home's temperature and humidity with the DHT11 sensor, accessible remotely through Arduino Cloud.

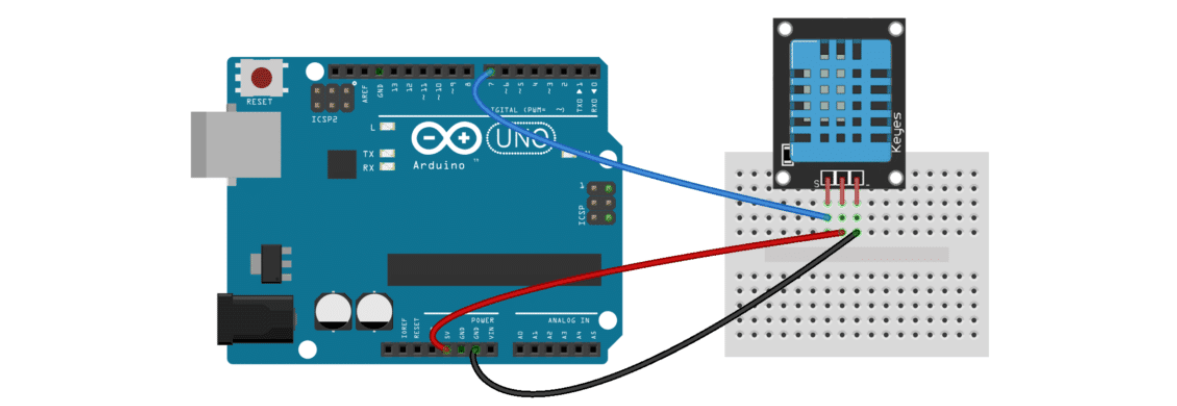
**Customizable ambiance:** Set the mood with the RGB LED and control individual blue, and red LEDs to create personalized lighting schemes.

# **Arduino Diagram**



**Thinkercad Link:** <https://www.tinkercad.com/things/3Prs292AaEp-iot-final-project>



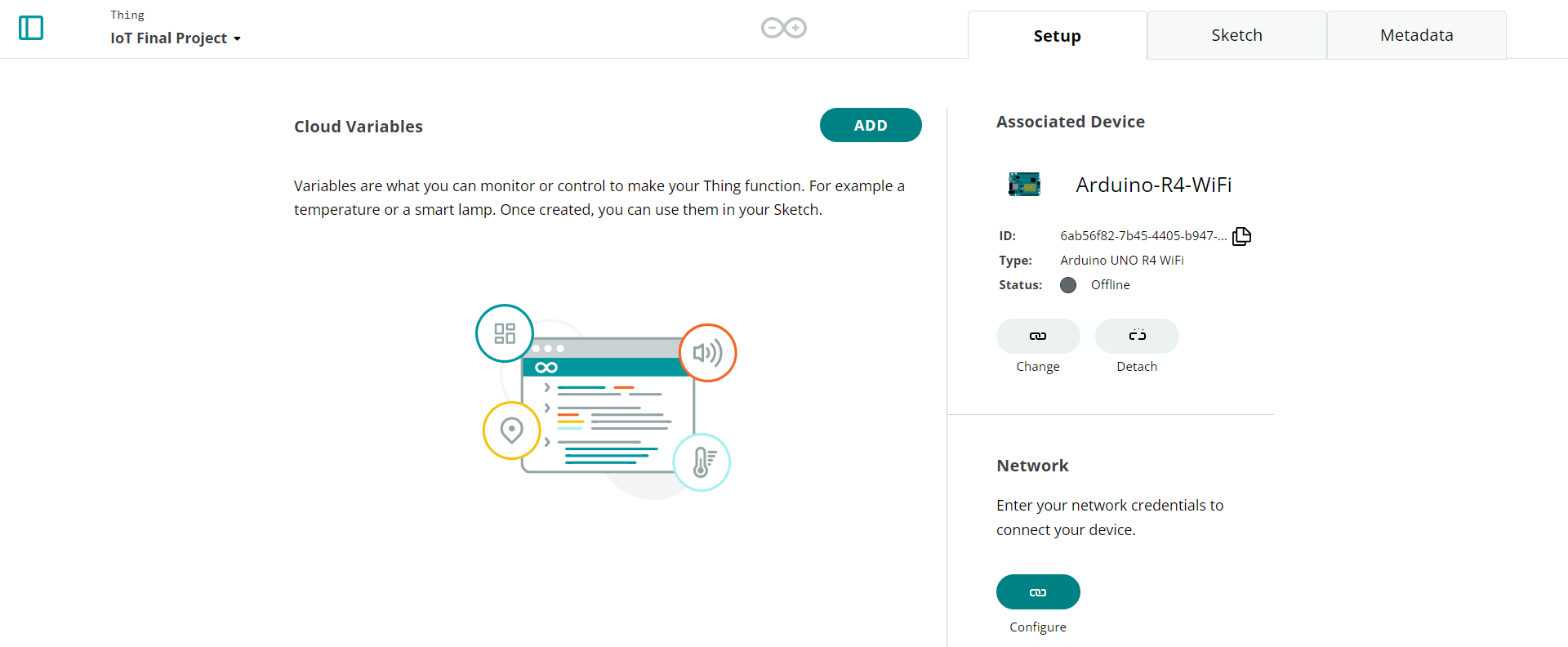


# **Components and supplies**

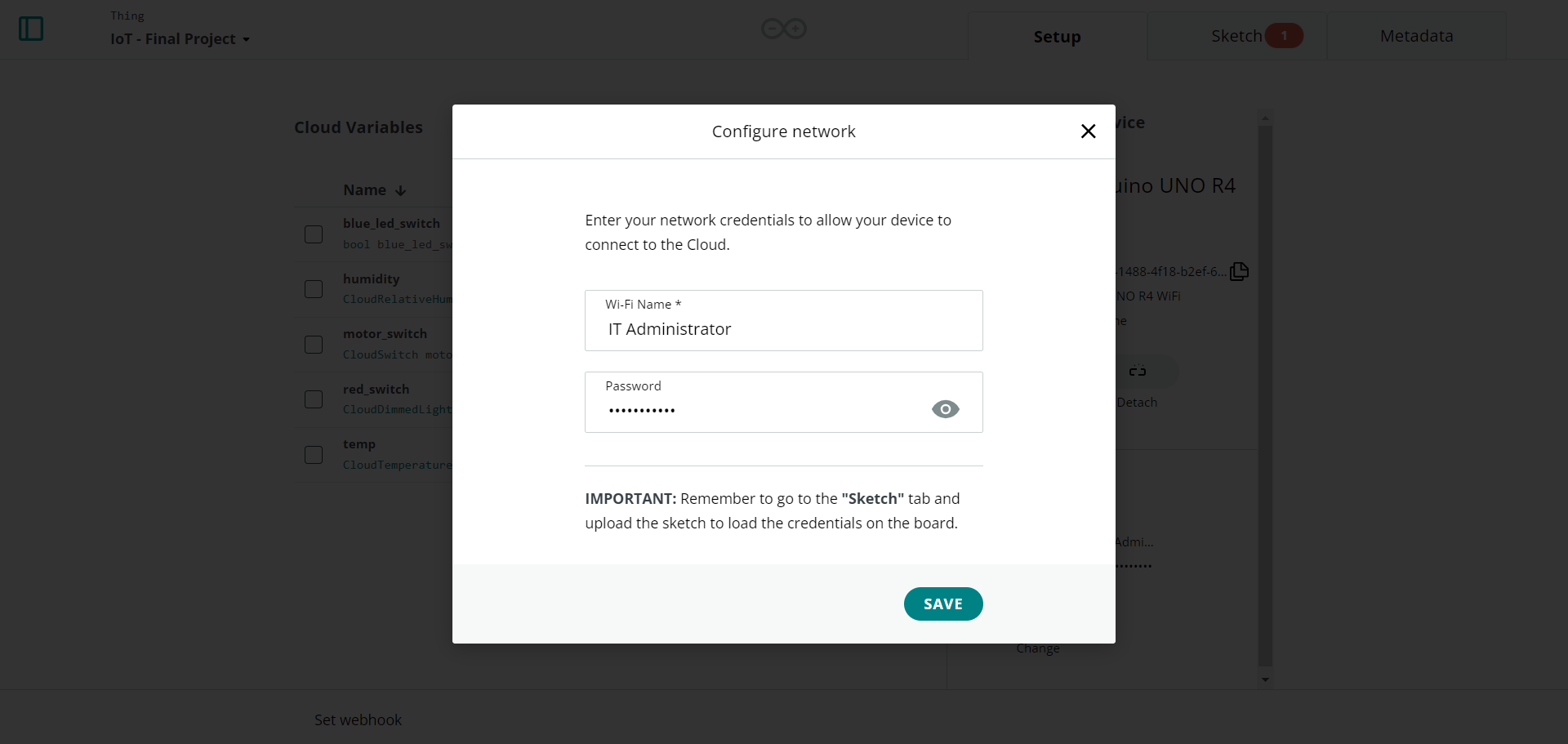
* Arduino UNO R4 WIFI
* IC2 Liquid Crystal Display
* DHT11
* 3 x 1K Ω
* 220 Ω
* RGB Led
* 4 x Blue Led, One White Led, and One Red Led
* HC-SR501 PIR Motion
* 330 Ω
* Green Led

# **Connecting Arduino UNO R4 WIFI to Arduino Cloud**

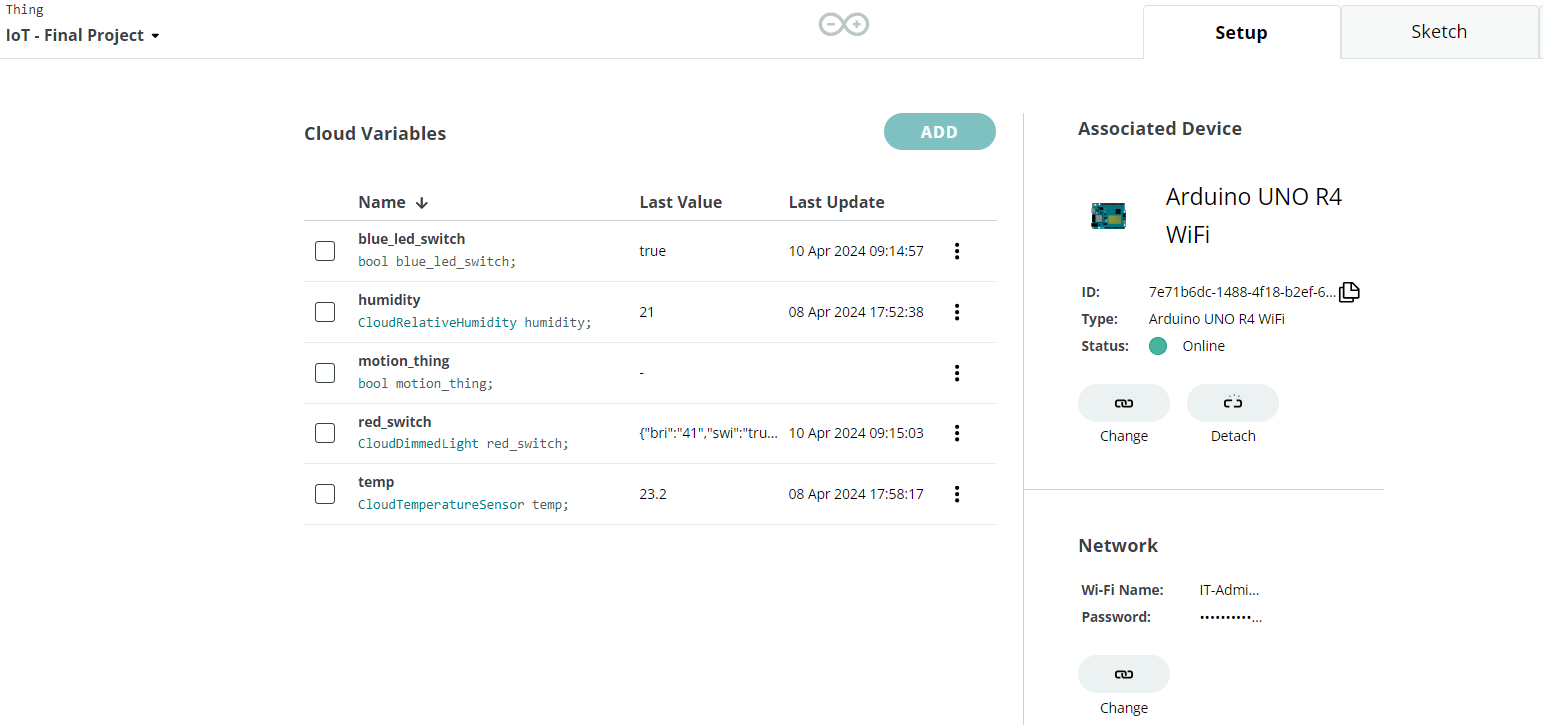
Once we have connected the Arduino to the PC and Arduino Cloud, we will create a "**Thing**" and then connect the Arduino to the network, finally, we can modify the sketch and add variables.



Make sure the Arduino UNO R4 WiFi is connected to our network:

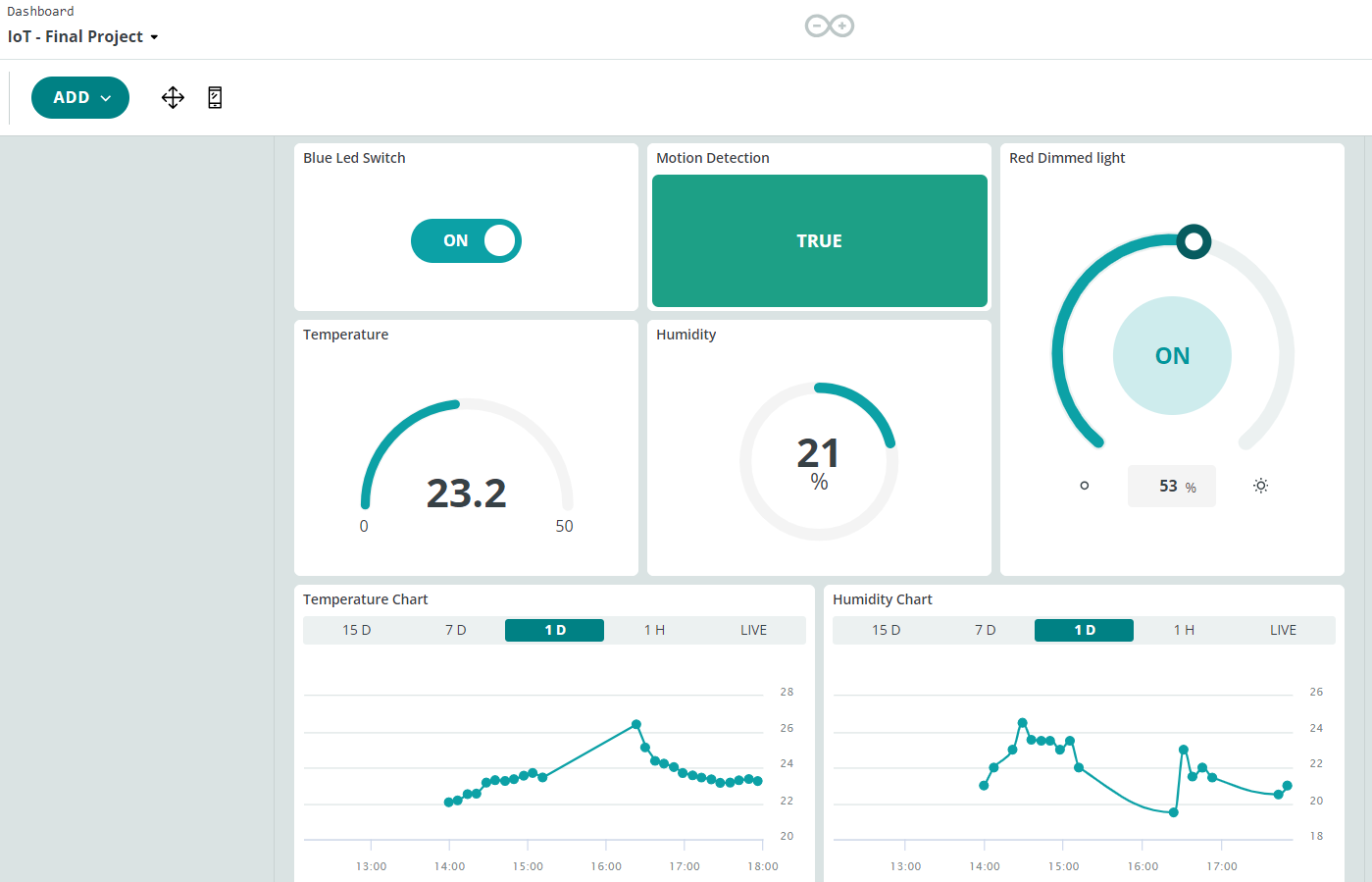


# **Creating the Thing and Dashboard**

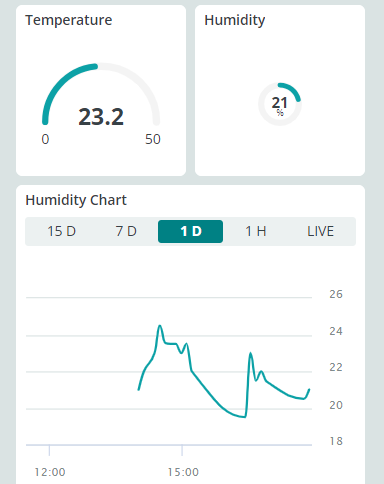
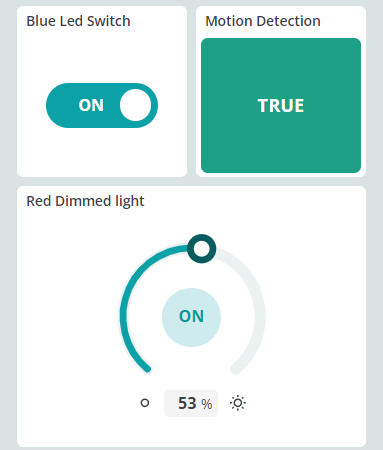
****

After adding the cloud variables and ensuring the status of the Arduino board is **Online**, only after that can we begin to create the dashboard for the Thing.

* **Desktop Layout:**



* **Mobile Layout:**

****

# **Arduino Code**

/\*

Sketch generated by the Arduino IoT Cloud Thing "Untitled"

https://create.arduino.cc/cloud/things/36ab5049-8455-48cd-82ed-8ea5e1ccd8b1

Arduino IoT Cloud Variables description

The following variables are automatically generated and updated when changes are made to the Thing

CloudDimmedLight red\_switch;

CloudTemperatureSensor temp;

CloudRelativeHumidity humidity;

bool blue\_led\_switch;

bool motion\_thing;

Variables which are marked as READ/WRITE in the Cloud Thing will also have functions

which are called when their values are changed from the Dashboard.

These functions are generated with the Thing and added at the end of this sketch.

\*/

#include <DHT.h>

#include <DHT\_U.h>

#include <LCD\_I2C.h>

#include "thingProperties.h"

#define dht\_sensor\_pin 13

#define dht\_sensor\_type DHT11

DHT dht\_sensor(dht\_sensor\_pin, dht\_sensor\_type);

LCD\_I2C lcd(0x27, 16, 2);

//PIR SENSOR

int pir\_sensor\_pin = 4;

int led\_pir = 3;

int val\_pir = 0;

//RGB LED, PWM LED AND BLUE LEDS

int LED\_PWM\_PIN = 6;

int led\_pin1b = 5;

int rgbR = 9;

int rgbG = 10;

int rgbB = 11;

void setup() {

// Initialize serial and wait for port to open:

Serial.begin(9600);

// This delay gives the chance to wait for a Serial Monitor without blocking if none is found

delay(1500);

// Defined in thingProperties.h

initProperties();

// Connect to Arduino IoT Cloud

ArduinoCloud.begin(ArduinoIoTPreferredConnection);

/\*

The following function allows you to obtain more information

related to the state of network and IoT Cloud connection and errors

the higher number the more granular information you’ll get.

The default is 0 (only errors).

Maximum is 4

\*/

setDebugMessageLevel(2);

ArduinoCloud.printDebugInfo();

pinMode(rgbR, OUTPUT);

pinMode(rgbG, OUTPUT);

pinMode(rgbB, OUTPUT);

pinMode(led\_pin1b, OUTPUT);

//PIR Setup

pinMode(led\_pir, OUTPUT);

pinMode(pir\_sensor\_pin, INPUT);

//DHT11 Setup

pinMode(dht\_sensor\_pin, INPUT);

dht\_sensor.begin();

lcd.begin(); // initialize the lcd

lcd.backlight();

}

void loop() {

ArduinoCloud.update();

val\_pir = digitalRead(pir\_sensor\_pin);

if(val\_pir == HIGH){

digitalWrite(led\_pir,HIGH);

if(motion\_thing == LOW){

Serial.println("Motion Detected");

motion\_thing == HIGH;

}

}else{

digitalWrite(led\_pir,LOW);

if(motion\_thing == HIGH){

Serial.println("Motion Ended, no detection.");

motion\_thing == LOW;

}

}

humidity = dht\_sensor.readHumidity();

temp = dht\_sensor.readTemperature();

if(isnan(humidity) || isnan(temp)){

Serial.println("Failed to read from DHT11 Sensor.");

}

else{

//DHT11 Information in the Serial Monitor

Serial.print("Humidity: ");

Serial.print(humidity);

Serial.print("%");

Serial.print(" | ");

//Temperature Data

Serial.print("Temperature: ");

Serial.print(temp);

Serial.println("°C");

//LCD Information

lcd.setCursor(0,0);

lcd.print("Temp: ");

lcd.print(temp);

lcd.print((char)223);

lcd.print("C");

lcd.setCursor(0,1);

lcd.print("Humidity: ");

lcd.print(humidity);

lcd.print("%");

}

delay(1250);

if(temp < 10){

analogWrite(rgbR, HIGH);

analogWrite(rgbG, HIGH);

analogWrite(rgbB, LOW);

}else if(10 <= temp && temp <= 25){

analogWrite(rgbR, LOW);

analogWrite(rgbG, HIGH);

analogWrite(rgbB, HIGH);

}

else{

analogWrite(rgbR, HIGH);

analogWrite(rgbG, LOW);

analogWrite(rgbB, HIGH);

}

}

/\*

Since BlueLedSwitch is READ\_WRITE variable, onBlueLedSwitchChange() is

executed every time a new value is received from IoT Cloud.

\*/

void onBlueLedSwitchChange() {

// Add your code here to act upon BlueLedSwitch change

if(blue\_led\_switch == true){

digitalWrite(led\_pin1b, HIGH);

}

else{

digitalWrite(led\_pin1b, LOW);

}

}

/\*

Since RedSwitch is READ\_WRITE variable, onRedSwitchChange() is

executed every time a new value is received from IoT Cloud.

\*/

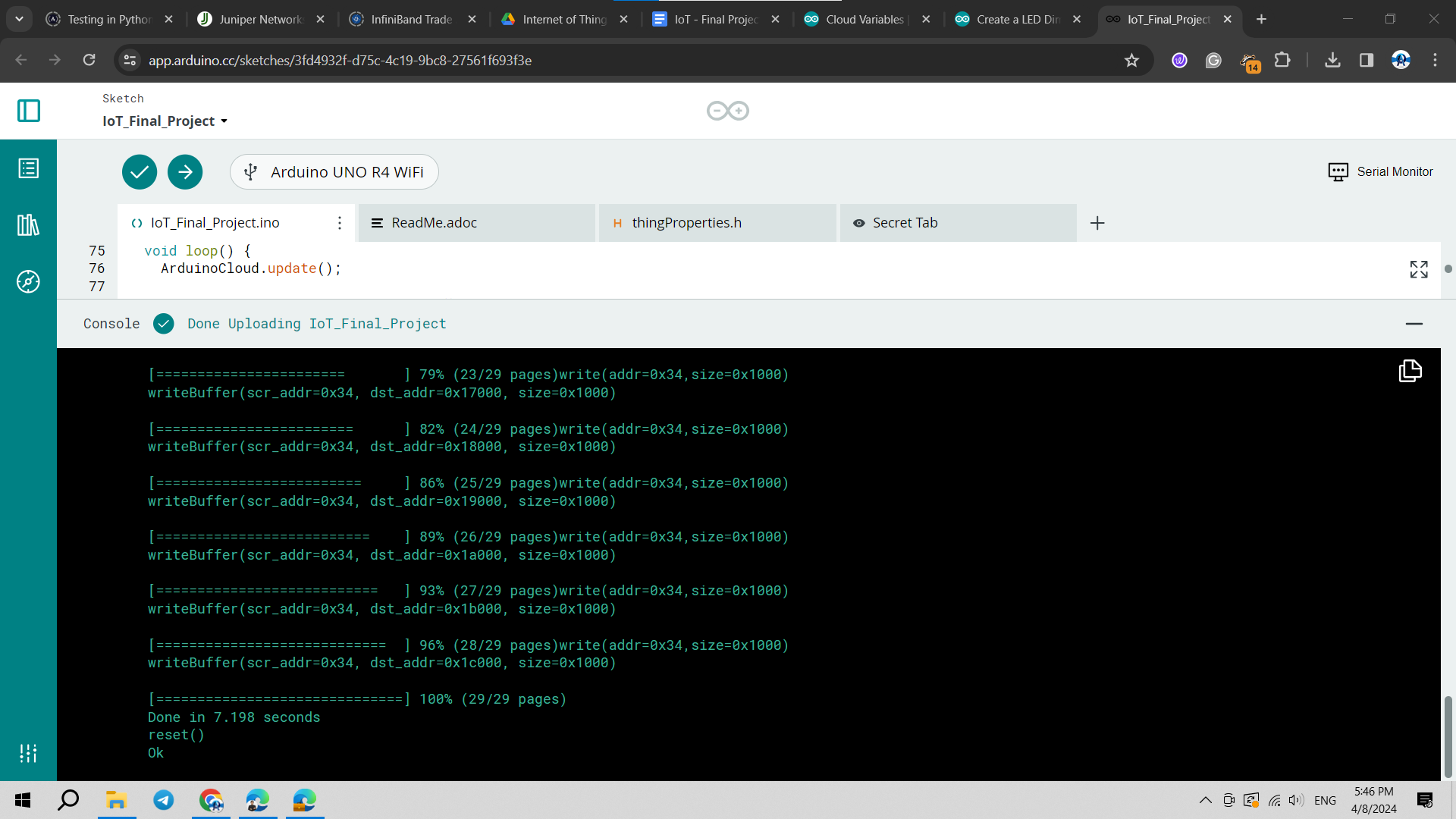
void onRedSwitchChange() {

int brightness = map(red\_switch.getBrightness(), 0, 100, 0, 255);

analogWrite(LED\_PWM\_PIN, brightness); // AnalogWrite for PWM control

}

Verifying and Uploading the Code on the Arduino UNO R4 WiFi:



# **References**

<https://cloud.arduino.cc/>

<https://docs.arduino.cc/arduino-cloud/cloud-interface/things/>

<https://docs.arduino.cc/arduino-cloud/cloud-interface/dashboard-widgets/>

<https://docs.arduino.cc/arduino-cloud/features/advanced-chart/>

<https://docs.arduino.cc/arduino-cloud/iot-remote-app/getting-started/>

<https://www.arduino.cc/reference/en/libraries/pir/>

<https://projecthub.arduino.cc/arduino_uno_guy/i2c-liquid-crystal-displays-5eb615>

<https://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/>

<https://docs.arduino.cc/arduino-cloud/cloud-interface/variables/>

<https://www.arduino.cc/reference/en/>